PROJECT #11: COMPUTATION OF LINEARIZED EULER EQUATIONS

This project solves the unsteady Euler equations for vortex propagation. The initial condition is a finite core vortex which will propagated from left to right at a fixed speed, M_∞. The domain is periodic in x, so boundary conditons a simple.

Equations: The Euler equations in two dimensions can be written as

$$\frac{\partial Q}{\partial t} + \frac{\partial E(Q)}{\partial x} + \frac{\partial F(Q)}{\partial y} = 0 \tag{1}$$

with

$$Q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ e \end{pmatrix}, E = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (e+p)u \end{pmatrix}, \quad F = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (e+p)v \end{pmatrix} \tag{2}$$

with the equation of state for pressure is $p = ((\gamma - 1))(e - \frac{1}{2}\rho(u^2 + v^2))$, temperature $T = \frac{p}{\rho}$ and $\gamma = 1.4$.

Note: Q, E, F are 4×1 vectors.

Geometry: The grid is uniform in $0 \leq x \leq 10$ and $0 \leq y \leq 10$.

Boundary conditions: Periodic in x and y. Note the domain is rectangular with limits $(0, 10)$, but the flow is assumed to be periodic over those limits.

Initial conditions and exact solution: A finite core vortex embedded in a free stream flow $Q_\infty = (\rho_\infty, M_\infty, 0, e_\infty)$ with $\rho_\infty = 1.0, M_\infty = 0., p_\infty = 1.0$, and $T_\infty = 1.0$. The perturbed (by the vortex) field is

$$T = T_\infty - \frac{\alpha^2(\gamma - 1)}{16\phi\gamma\pi^2} e^{2\phi(1-r^2)} \tag{3}$$

$$\rho = T^{\frac{1}{(\gamma-1)}} \tag{4}$$

$$u = M_\infty - \frac{\alpha}{2\pi}(y - y_0)e^{\phi(1-r^2)} \tag{5}$$

$$v = \frac{\alpha}{2\pi}(x - x_0)e^{\phi(1-r^2)} \tag{6}$$

with α the vortex strength (use $\alpha = 5.0$) and $\phi = 0.5$ is the gaussian width scale. The vortex is initally centered at $x_0 = 5.0$ and $y_0 = 5.0$ and $r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$

Project 11 Apply the Euler Explicit and RK4 Time Differencing and Flux Splitting Method in space.

First starting with, the Euler explicit time differening ($h = \Delta t$), we have

$$Q^{(n+1)} = Q^{(n)} + hR(Q)^{(n)} \tag{7}$$

where

$$R(Q)^{(n)} = -\frac{\partial E^n}{\partial x} - \frac{\partial F^n}{\partial y} \tag{8}$$

Using nx, the number of points in x and y (keep the aspect ratio square), let $\Delta x = \Delta y = \frac{10.0}{nx-1}$. Discretize the field using a uniform grid with $x_{j,k} = (j-1)\Delta x$ and $y_{j,k} = (k-1)\Delta y$, for $j = 1, 2, 3, \ldots nx-1$ and $k = 1, 2, 3, \ldots nx-1$. Note the computational domain falls short of 10 by Δx and Δy, in x and y respectively, since e.g. $x = 0$ and $x = 10.0$ are the same location because of the periodicity conditions.

The flux Jacobians are $A = \frac{\partial E}{\partial Q}$ and $B = \frac{\partial F}{\partial Q}$. The matries A, B can be $\pm$ flux split into A^+, B^+ and A^-, B^- as discussed in class. New split fluxes are defined as $E^\pm = A^\pm Q$ and $F^\pm = B^\pm Q$. This produces the new system to be solved with

$$R(Q)_{j,k}^{(n)} = -\delta_x^b (E^+)_{j,k}^{(n)} - \delta_x^f (E^-)_{j,k}^{(n)} - \delta_y^b (F^+)_{j,k}^{(n)} - \delta_y^f (F^-)_{j,k}^{(n)} \tag{9}$$

where e.g., δ_x^b is a backward differencing operator and δ_x^f is a forward differencing operator.

Sample Project Code

A sample Matlab code is provided for you. It uses Euler explicit time differencing with central space differencing. It is unstable for all $CFL = \frac{\Delta t M_\infty}{\Delta x}$, but will run for awhile at low CFL. Try $CFL = 0.1$ and $nx = 21$. I suggest you use this code as a starting point. It produces plots of the vortex as it propagates from left to right, through the right boundary reappearing at the left boundary and continuing on. In fact, I set it up so that you can pick the number of revolutions of the vortex and for an inputted CFL compute the number of time steps to complete the revolutions.

Assignment

1. Program the Euler Explicit scheme Eq. 7 with the Flux Split form, Eq. 9. Use 1^{st},2^{nd} and 3^{rd} order one-sided differences for $\delta_x^b, \delta_x^f, \delta_y^b$, and δ_y^f.

2. Replace the Euler Explicit scheme with Fourth-Order Runge-Kutta and compare the performance and other aspects discussed below.

   (a) The RK4 scheme is defined as

$$\begin{aligned}
\widehat{Q}^{n+1/2} &= Q^n + \tfrac{1}{2} h R(Q)^n \\
\tilde{Q}^{n+1/2} &= Q^n + \tfrac{1}{2} h R(\widehat{Q})^{n+1/2} \\
\overline{Q}^{n+1} &= Q^n + h R(\tilde{Q})^{n+1/2} \\
Q^{n+1} &= Q^n + \tfrac{1}{6} h \left[R(Q)^n + 2\left(R(\widehat{Q})^{n+1/2} + R(\tilde{Q})^{n+1/2} \right) + R(\overline{Q})^{n+1} \right]
\end{aligned} \tag{10}$$

   (b) **Note: the proper use of $\widehat{Q}, \tilde{Q}, \overline{Q}$ is very important!**

3. Discuss the spatial accuracy of this method, e.g., what is er_t for the various differences or what is the modified wave number? *Hint* Remember, we are working with a non-linear coupled system. To do the analysis you need to think in terms of the decouple representative equations.

4. Knowing the $\sigma - \lambda$ relation for the Euler explicit and RK4 $O\Delta E$, come up with a stability condition and convergence estimates for different differencing orders. One possible CFL definition for this system is

$$CFL = \frac{h \; max(\lambda(A), \lambda(B))}{\min(dx, dy)}$$

5. Study various CFL numbers (remembering the numerical stability condition!), to see if your analysis is consistent with your results.

6. Pick one:

   (a) Do a grid refinement accuracy study to verify the truncation error for each difference choice, .i.e. $1^{st}, 2^{nd}$ and 3^{rd} Order upwind. Set up a sequence of runs with decreasing grid (increasing nx) and time step sizes. The best way to do this is to intergrate to fixed time, say one revolution. Keep in mind you should keep the CFL number fixed, that is, as you decrease Δx you should correspondingly decrease Δt. Plot the $log(error)$ ($error$ can be defined as the $l2$ norm of the difference between the computed density and the exact density) against $log(\Delta x)$. If the $error \approx O(\Delta x^p)$ then, for example, on a loglog plot a $O(\Delta x^p)$ method should have a slope of p. Verify this for your results.

   (b) Program up at least one more method $O\Delta E$ method and compare it's performance, accuracy and stability with your results above. Look at the other projects for suggestions or just pick one from the textbook or other sources.

      i. *Note:* It is not necessary to redo the accuracy and stability analysis for this method, although you may want to make sure it is stable and accurate.)

      ii. *Suggestion:* Take a look at Lax-Wendroff (see notes). You could even use an unstable scheme, although I prefer a stable one.

   (c) Program the Euler Implicit scheme and compare it's efficiency with the Euler explicit results. You can do this with a central difference implicit operator or an LU operator. See the Linear Euler Equation Project #9, for more details.

Suggestions, Questions:

1. You can play around with the value of M_∞ or some of the other parameters.

2. You should play around with the CFL. Why? Is there an optimal value of CFL for accurate results and the best convergence?

3. What happens when you violate the stability condition? Try it.

General Instructions:
Follow the instructions given above and address each of the assignments. You will need to provide me with a **short** writeup of what you have done, along with some results and

figures. This can be handwritten, but I prefer TeX, LaTeX or some other word processor form. Perform all the computations using MATLAB. I will also want copies of all the source codes. (You will be required to email them to me, I will make arrangements).
This project will account for 50% of your grade. I will be judging it on the write-up, and a working code (I will run all the codes you send me).